Project: No-show appointments

Table of Contents

Introduction

> This dataset collects information from 100k medical appointments in Brazil and is focused on the question of whether or not patients show up for their appointment.

In [165…

```
#import libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline
```

Data Wrangling

General Properties

Read data file

In [136…

```
df=pd.read_csv(r"C:\Users\HP\.jupyter\noshowappointments-kagglev2-may-2016.csv")
df
```

Out[136]:

	PatientId	AppointmentID	Gender	ScheduledDay	AppointmentDay	Age	Neighbourhood	Scholarsh
0	2.987250e+13	5642903	F	2016-04-29T18:38:08Z	2016-04-29T00:00:00Z	62	JARDIM DA PENHA	
1	5.589978e+14	5642503	M	2016-04-29T16:08:27Z	2016-04-29T00:00:00Z	56	JARDIM DA PENHA	
2	4.262962e+12	5642549	F	2016-04-29T16:19:04Z	2016-04-29T00:00:00Z	62	MATA DA PRAIA	
3	8.679512e+11	5642828	F	2016-04-29T17:29:31Z	2016-04-29T00:00:00Z	8	PONTAL DE CAMBURI	
4	8.841186e+12	5642494	F	2016-04-29T16:07:23Z	2016-04-29T00:00:00Z	56	JARDIM DA PENHA	
...	...	...	...	...	...	...	...	
110522	2.572134e+12	5651768	F	2016-05-	2016-06-	56	MARIA ORTIZ	

				03T09:15:35Z	07T00:00:00Z		
110523	3.596266e+12	5650093	F	2016-05-03T07:27:33Z	2016-06-07T00:00:00Z	51	MARIA ORTIZ
110524	1.557663e+13	5630692	F	2016-04-27T16:03:52Z	2016-06-07T00:00:00Z	21	MARIA ORTIZ
110525	9.213493e+13	5630323	F	2016-04-27T15:09:23Z	2016-06-07T00:00:00Z	38	MARIA ORTIZ
110526	3.775115e+14	5629448	F	2016-04-27T13:30:56Z	2016-06-07T00:00:00Z	54	MARIA ORTIZ

110527 rows × 14 columns

Number of rows and columns in dataset

In [137…

```
df.shape
#number raws,number columns
```

Out[137]:

```
(110527, 14)
```

Describe the dataset

In [138…

```
df.describe()
```

Out[138]:

	PatientId	AppointmentID	Age	Scholarship	Hipertension	Diabetes	Alcoholism
count	1.105270e+05	1.105270e+05	110527.000000	110527.000000	110527.000000	110527.000000	110527.000000
mean	1.474963e+14	5.675305e+06	37.088874	0.098266	0.197246	0.071865	0.030400
std	2.560949e+14	7.129575e+04	23.110205	0.297675	0.397921	0.258265	0.171686
min	3.921784e+04	5.030230e+06	-1.000000	0.000000	0.000000	0.000000	0.000000
25%	4.172614e+12	5.640286e+06	18.000000	0.000000	0.000000	0.000000	0.000000
50%	3.173184e+13	5.680573e+06	37.000000	0.000000	0.000000	0.000000	0.000000
75%	9.439172e+13	5.725524e+06	55.000000	0.000000	0.000000	0.000000	0.000000
max	9.999816e+14	5.790484e+06	115.000000	1.000000	1.000000	1.000000	1.000000

Info about dataset

In [139…

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 110527 entries, 0 to 110526
Data columns (total 14 columns):
 #   Column          Non-Null Count   Dtype  
---  ------          --------------   -----  
 0   PatientId       110527 non-null  float64
 1   AppointmentID   110527 non-null  int64  
 2   Gender          110527 non-null  object 
 3   ScheduledDay    110527 non-null  object 
 4   AppointmentDay  110527 non-null  object 
 5   Age             110527 non-null  int64  
 6   Neighbourhood   110527 non-null  object 
 7   Scholarship     110527 non-null  int64  
 8   Hipertension    110527 non-null  int64
 9   Diabetes        110527 non-null  int64
 10  Alcoholism      110527 non-null  int64
 11  Handcap         110527 non-null  int64
 12  SMS_received    110527 non-null  int64
 13  No-show         110527 non-null  object
dtypes: float64(1), int64(8), object(5)
memory usage: 11.8+ MB
```

convert data type in ScheduledDay AppointmentDay to datetime

Number of nulls in each column

In [140…

```
df.isnull().sum()
```

Out[140]:

```
PatientId         0
AppointmentID     0
Gender            0
ScheduledDay      0
AppointmentDay    0
Age               0
Neighbourhood     0
Scholarship       0
Hipertension      0
Diabetes          0
Alcoholism        0
Handcap           0
SMS_received      0
No-show           0
dtype: int64
```

there is no missing value

Number of unique in each column

In [141…

```
df.nunique()
```

Out[141]:

```
PatientId          62299
AppointmentID     110527
Gender                 2
ScheduledDay      103549
AppointmentDay        27
Age                  104
Neighbourhood         81
Scholarship            2
Hipertension           2
Diabetes               2
Alcoholism             2
Handcap                5
SMS_received           2
No-show                2
dtype: int64
```

will drop PatientId and AppointmentID

Number of duplicates in dataset

In [142…

```
df.duplicated().sum()
```

Out[142]:

```
0
```

will drop these duplicates

Number of ages less than 0

In [143…

```
df.query('Age <0')
```

Out[143]:

	PatientId	AppointmentID	Gender	ScheduledDay	AppointmentDay	Age	Neighbourhood	Scholarshi
99832	4.659432e+14	5775010	F	2016-06-06T08:58:13Z	2016-06-06T00:00:00Z	-1	ROMÃO	

Dataframe for show up

In [144…

```
df_No=df[df['No-show']=='No']
df_No.head()
```

Out[144]:

	PatientId	AppointmentID	Gender	ScheduledDay	AppointmentDay	Age	Neighbourhood	Scholarship	H
0	2.987250e+13	5642903	F	2016-04-29T18:38:08Z	2016-04-29T00:00:00Z	62	JARDIM DA PENHA	0	
1	5.589978e+14	5642503	M	2016-04-29T16:08:27Z	2016-04-29T00:00:00Z	56	JARDIM DA PENHA	0	
2	4.262962e+12	5642549	F	2016-04-29T16:19:04Z	2016-04-29T00:00:00Z	62	MATA DA PRAIA	0	
3	8.679512e+11	5642828	F	2016-04-29T17:29:31Z	2016-04-29T00:00:00Z	8	PONTAL DE CAMBURI	0	
4	8.841186e+12	5642494	F	2016-04-29T16:07:23Z	2016-04-29T00:00:00Z	56	JARDIM DA PENHA	0	

Dataframe for no-show

In [145…

```
df_Yes=df[df['No-show']=='Yes']
df_Yes.head()
```

Out[145]:

	PatientId	AppointmentID	Gender	ScheduledDay	AppointmentDay	Age	Neighbourhood	Scholarship
6	7.336882e+14	5630279	F	2016-04-27T15:05:12Z	2016-04-29T00:00:00Z	23	GOIABEIRAS	0
7	3.449833e+12	5630575	F	2016-04-27T15:39:58Z	2016-04-29T00:00:00Z	39	GOIABEIRAS	0
11	7.542951e+12	5620163	M	2016-04-26T08:44:12Z	2016-04-29T00:00:00Z	29	NOVA PALESTINA	0
17	1.479497e+13	5633460	F	2016-04-28T09:28:57Z	2016-04-29T00:00:00Z	40	CONQUISTA	1
20	6.222575e+14	5626083	F	2016-04-27T07:51:14Z	2016-04-29T00:00:00Z	30	NOVA PALESTINA	0

Data Cleaning

drop the columns not I need

In [146…

```
df.drop(['PatientId','AppointmentID'],axis=1,inplace=True)
df
```

Out[146]:

	Gender	ScheduledDay	AppointmentDay	Age	Neighbourhood	Scholarship	Hipertension	Diabetes	Al
0	F	2016-04-29T18:38:08Z	2016-04-29T00:00:00Z	62	JARDIM DA PENHA	0	1	0	
1	M	2016-04-29T16:08:27Z	2016-04-29T00:00:00Z	56	JARDIM DA PENHA	0	0	0	
2	F	2016-04-29T16:19:04Z	2016-04-29T00:00:00Z	62	MATA DA PRAIA	0	0	0	
3	F	2016-04-29T17:29:31Z	2016-04-29T00:00:00Z	8	PONTAL DE CAMBURI	0	0	0	
4	F	2016-04-29T16:07:23Z	2016-04-29T00:00:00Z	56	JARDIM DA PENHA	0	1	1	
...	...	...	...	...	...	...	...	...	
110522	F	2016-05-03T09:15:35Z	2016-06-07T00:00:00Z	56	MARIA ORTIZ	0	0	0	
110523	F	2016-05-03T07:27:33Z	2016-06-07T00:00:00Z	51	MARIA ORTIZ	0	0	0	
110524	F	2016-04-27T16:03:52Z	2016-06-07T00:00:00Z	21	MARIA ORTIZ	0	0	0	
110525	F	2016-04-27T15:09:23Z	2016-06-07T00:00:00Z	38	MARIA ORTIZ	0	0	0	
110526	F	2016-04-27T13:30:56Z	2016-06-07T00:00:00Z	54	MARIA ORTIZ	0	0	0	

110527 rows × 12 columns

drop of duplicates

In [147…

```
df.drop_duplicates(inplace=True)
```

In [148…

```
df.duplicated().sum()
```

Out[148]:

```
0
```

Drop age less than 0

In [149…

```
df.drop(99832,inplace=True)
```

In [150…

```
#make sure to delelte
df.query('Age <0')
```

Out[150]:

Gender	ScheduledDay	AppointmentDay	Age	Neighbourhood	Scholarship	Hipertension	Diabetes	Alcoholisr

Convert these columns to datetime

In [151…

```
df["AppointmentDay"]=pd.to_datetime(df["AppointmentDay"])
df["ScheduledDay"]=pd.to_datetime(df["ScheduledDay"])
```

In [161…

```
df.info()
```

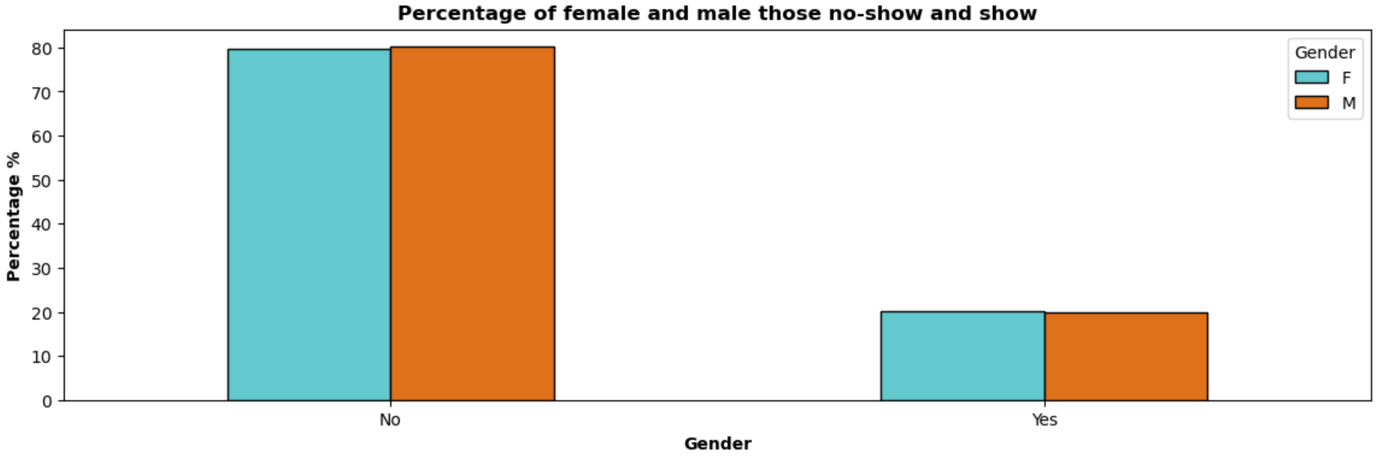
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 109891 entries, 0 to 110526
Data columns (total 12 columns):
 #   Column          Non-Null Count   Dtype
---  ------          --------------   -----
 0   Gender          109891 non-null  object
 1   ScheduledDay    109891 non-null  datetime64[ns, UTC]
 2   AppointmentDay  109891 non-null  datetime64[ns, UTC]
 3   Age             109891 non-null  int64
 4   Neighbourhood   109891 non-null  object
 5   Scholarship     109891 non-null  int64
 6   Hipertension    109891 non-null  int64
 7   Diabetes        109891 non-null  int64
 8   Alcoholism      109891 non-null  int64
 9   Handcap         109891 non-null  int64
 10  SMS_received    109891 non-null  int64
 11  No-show         109891 non-null  object
dtypes: datetime64[ns, UTC](2), int64(7), object(3)
memory usage: 10.9+ MB
```

Exploratory Data Analysis

What is percentage of female and male those no-show and show?

In [152…

```
g_no_show=df.groupby('Gender')['No-show'].value_counts(normalize=True).mul(100).unstack(
g_no_show.plot(kind='bar',edgecolor='black',rot=0,figsize=[14,4],color=['#64C9CF','#DF71
plt.title("Percentage of female and male those no-show and show",weight='bold')
plt.ylabel("Percentage %",weight='bold')
plt.xlabel("Gender",weight='bold');
```



- **This visualization shows that 80% of female and male attended their appointments and 20% of female and male not attended their appointments.**

What is the average age of those who missed their appointments?

In [153…

```
N_age_not=round(df_Yes.Age.mean())
N_age_not
```

Out[153]:

```
34
```

- **Mean age of those who missed their appointments: 34**

What is the average age of those who attend their appointments?

In [154…

```
df_No=df[df['No-show']=='No']
N_age=round(df_No.Age.mean())
N_age
```

Out[154]:

```
38
```

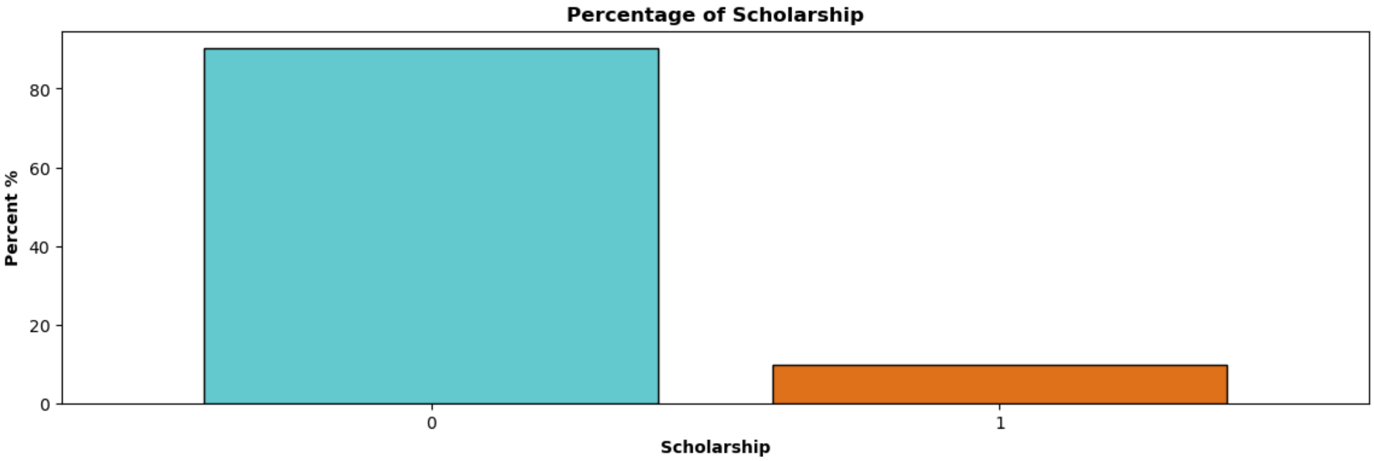
- **Mean age of those who attended their appointments: 38**

What percentage of scholarship?

In [162…

```
def myPlot(df,x):
    df[x].value_counts(normalize=True).mul(100).plot.bar(edgecolor='black',figsize=[14,4
    plt.title(f'Percentage of {x}',weight='bold')
    plt.xlabel(x.title(),weight='bold')
    plt.ylabel('Percent %',weight='bold')

myPlot(df,'Scholarship')
```

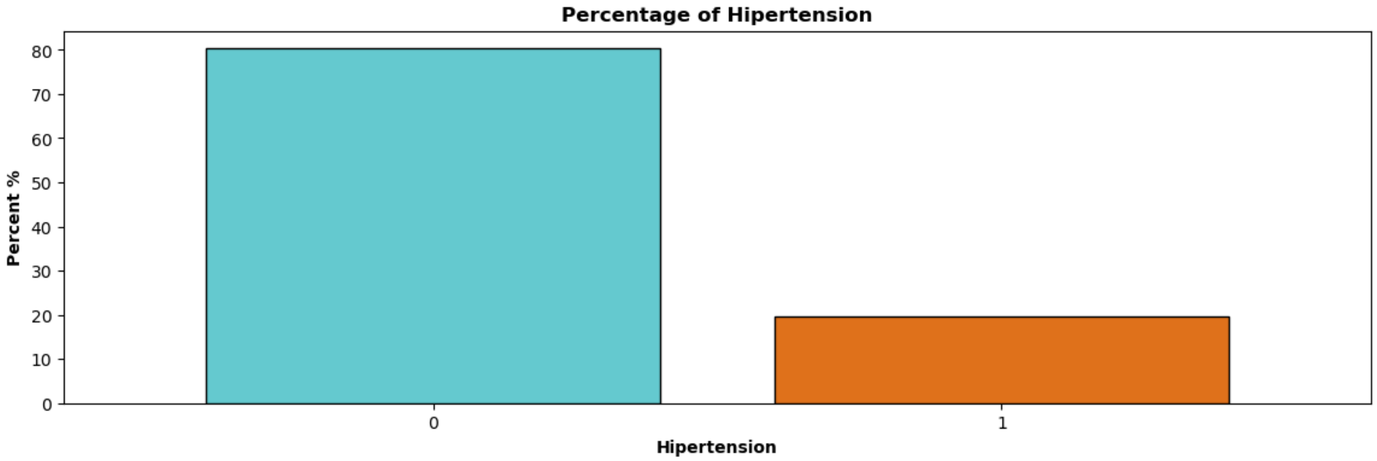


- **Percentage enrolled in scholarship is less than 10%**
- **Percentage not enrolled in scholarship is greater than 90%**

What percentage of Hipertension?

In [156…

```
myPlot(df,'Hipertension')
```

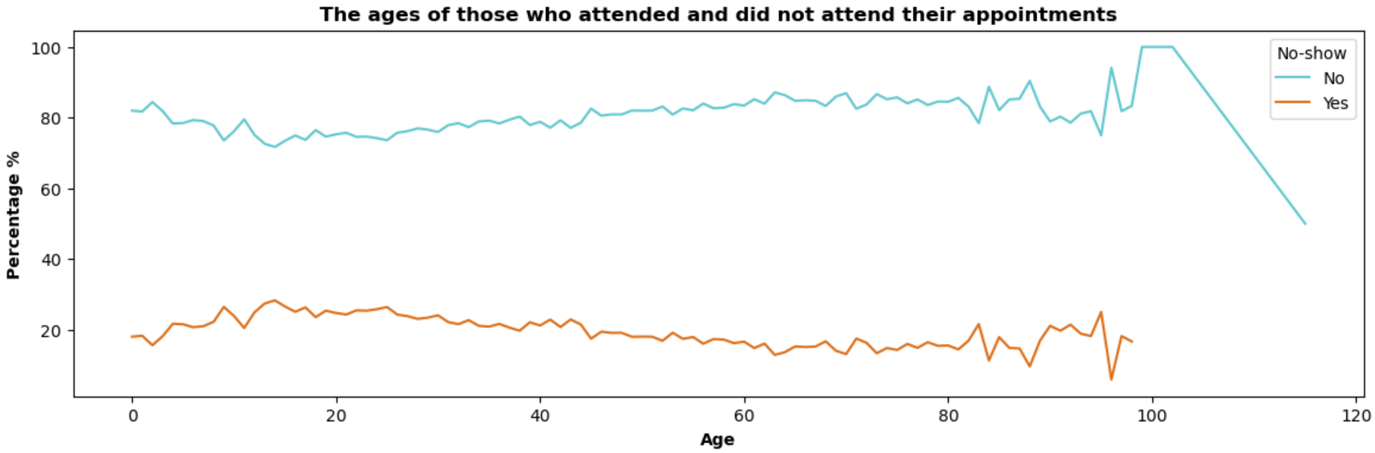


- More than 20% of patients have hypertension

What are the ages of those who attended and did not attend their appointments?

In [157…

```
df.groupby('Age')['No-show'].value_counts(normalize=True).mul(100).unstack('No-show').pl
plt.title('The ages of those who attended and did not attend their appointments',weight=
plt.xlabel("Age",weight='bold')
plt.ylabel("Percentage %",weight='bold');
```



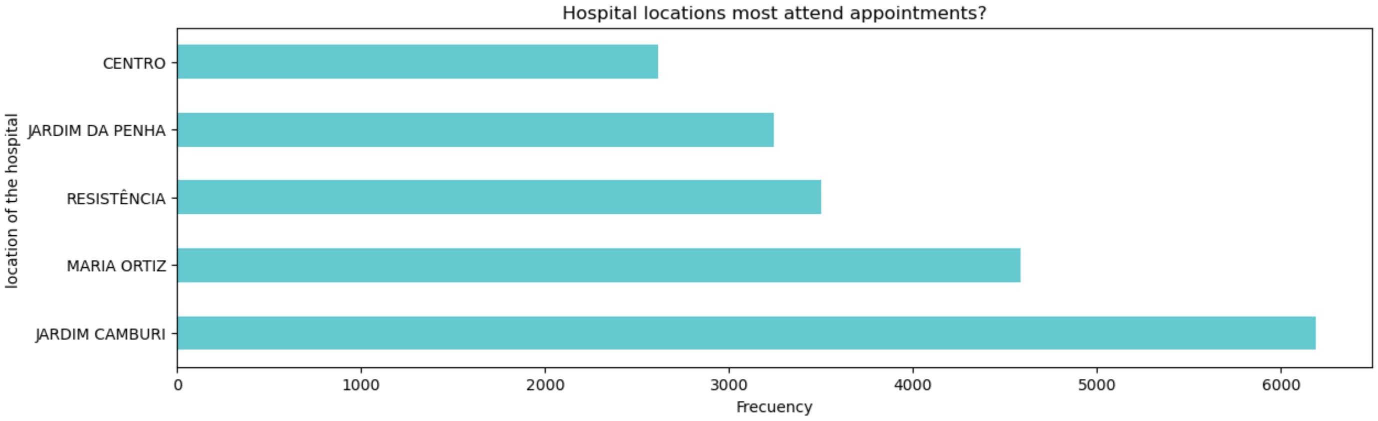
- We see an increase in the percentage of attendance of appointments from 20 to 80 years

and a decrease in the percentage of no-show from 20 to 80 years

What hospital locations most attend appointments?

In [158…

```
ne=df_No.Neighbourhood.value_counts()
plt.figure(figsize=[14,4])
ne[:5].plot(kind='barh',color='#64C9CF')
plt.title("Hospital locations most attend appointments?")
plt.xlabel('Frecuency')
plt.ylabel("location of the hospital ");
```



- **See top 5 hospital locations with attending appointments ,JARDIM CAMBURI with over 6000 appointment and MARIA ORTEZ with over 4,000 appointments.**

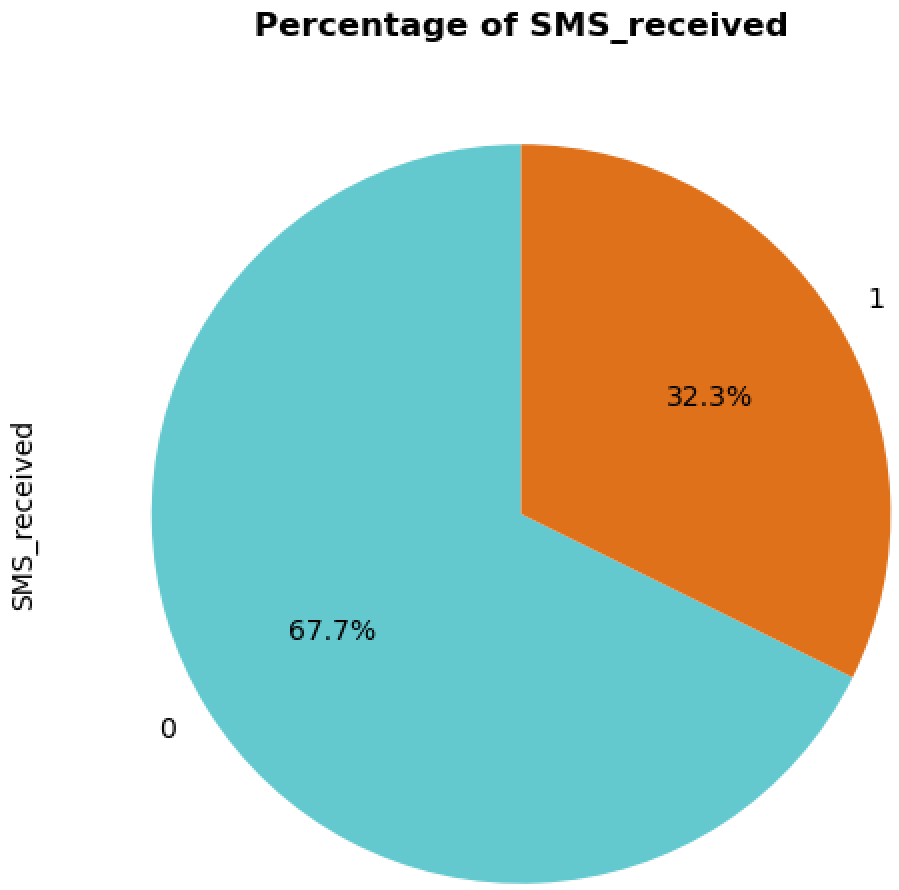
What percentage of SMS_received?

In [159…

```
c=['#64C9CF','#DF711B']
def myPlot(df,x):

    df[x].value_counts(normalize=True).mul(100).plot.pie(colors=c,figsize=(8,6),autopct=
    plt.title(f'Percentage of {x}',weight='bold');


myPlot(df,'SMS_received');
```



- we see more than 65% not received SMS

Conclusions

Limitations:

- The data contains approximately two months of appointments, and this is a short period of time.

- Values in Handcap are unclear.

- Most values contain categorical data.